# A Cloud-Based Real-Time Public Transport Tracking System Using Java Full-Stack Architecture

**Ms. Rajnandini Niloba Gidde[1], Ms. Dhanashri Mahadev Dhaygude[2], Ms. Divya Rajendra Ingale[3]**

[1,2,3]B. Tech Final Year, Computer Science & Engineering, Karmaveer Bhaurao Patil Colege of Engineering, Satara

**Abstract**

Urban public transportation systems face persistent challenges related to unpredictability, delays, and lack of real- time visibility for passengers. Traditional timetable-based systems often fail to provide accurate arrival information under dynamic traffic conditions, resulting in passenger dissatisfaction and inefficient transport operations. This paper presents the design and implementation of a cloud-based real-time public transport tracking system developed using a Java full-stack architecture. The proposed system enables live vehicle tracking, dynamic estimated time of arrival (ETA) prediction, real-time passenger notifications, and centralized route management. By leveraging RESTful services, WebSocket-based communication, and scalable cloud deployment, the system ensures low-latency updates and high availability. Experimental evaluation shows improved arrival time accuracy and enhanced user experience compared to static schedule-based approaches.

**Keyword:** Real-Time Tracking, Public Transport, Java Full-Stack, WebSockets, Cloud Computing, ETA Prediction

## I. Introduction

Rapid urbanization has led to a significant increase in the usage of public transportation systems. Despite this growth, passengers frequently experience uncertainty due to delays, traffic congestion, and lack of timely and accurate information. Conventional public transport systems mainly rely on fixed schedules, which do not adapt effectively to real-world traffic conditions.

With advancements in cloud computing, real-time commu- nication technologies, and web-based platforms, it has become possible to provide continuous updates and predictive insights to commuters. Real-time tracking systems help improve pas- senger satisfaction, reduce waiting times, and assist transport authorities in managing operations more efficiently.

This paper proposes a Real-Time Public Transport Tracking System (RPTTS) built using a Java full-stack architecture. The system offers live vehicle location tracking, accurate ETA calculation, and real-time notifications through a scalable and secure cloud-based infrastructure.

## II. Related Work

Several research studies have explored GPS-based track- ing systems for public transportation. Early implementations mainly focused on basic vehicle location visualization with- out real-time interaction or predictive capabilities. Although useful, these systems lacked responsiveness and scalability.

Recent solutions have introduced mobile applications and cloud services for real-time tracking. However, many rely heavily on frequent client-side polling, which increases server load and network traffic. In addition, some proprietary systems are costly and difficult to integrate with existing transport infrastructure.

The proposed system addresses these limitations by using WebSocket-based server push mechanisms for real-time up- dates. This approach minimizes latency and network overhead while enabling continuous live tracking. The modular Java full-stack design further ensures flexibility, scalability, and cost-effectiveness.

## III. System Architecture

The proposed system follows a layered and modular archi- tecture.

### A. *Data Acquisition Layer*

This layer is responsible for collecting vehicle location data from GPS-enabled devices or simulated data sources. Each vehicle periodically sends latitude, longitude, speed, and timestamp information to the backend server.

### B. *Backend Services Layer*

The backend is implemented using Spring Boot and exposes RESTful APIs for user authentication, route management, and administrative operations. Real-time vehicle updates are handled using Spring WebSocket, enabling bidirectional com- munication between the server and connected clients.

### C. *ETA Prediction Engine*

The ETA prediction engine calculates estimated arrival times using the current vehicle speed, distance to the next stop, and historical travel data. The engine dynamically updates predictions when delays or traffic variations are detected.

### D. *Frontend Layer*

The frontend provides an interactive web interface with real- time maps, route visualization, and notification alerts. Users can search routes, track vehicles live, and receive arrival or delay notifications.

### E. *Cloud Deployment Layer*

The system is containerized using Docker and deployed on cloud infrastructure such as AWS or Azure. This ensures scalability, fault tolerance, and high availability under varying user loads.

## IV. Implementation Details

### A. *Technology Stack*

The system is developed using the following technologies:

- Backend: Java 17, Spring Boot, Spring WebSocket, Spring Security
- Frontend: React, HTML5, CSS, JavaScript
- Database: MongoDB
- Security: JWT-based authentication and role-based access control
- APIs and Services: Google Maps Platform APIs
- Deployment: Docker and cloud platforms

### B. *Security Mechanisms*

Security is implemented using JSON Web Tokens (JWT) to enable stateless authentication. Role-based access control en- sures that only authorized administrators can perform sensitive operations such as route modification and system monitoring.

### C. *Real-Time Communication*

WebSockets are used to push real-time location and ETA up- dates to connected clients. This approach eliminates frequent polling and significantly reduces server load while maintaining low-latency communication.

## V. Experimental Evaluation

The system was evaluated under simulated urban traffic conditions involving multiple vehicles and concurrent users. Performance metrics considered include location update la- tency, ETA accuracy, and server response time.

Experimental results indicate that the system delivers real- time updates within a few seconds. ETA accuracy improved by approximately 20–30% compared to static schedule-based sys- tems. The WebSocket-based architecture demonstrated lower network overhead and improved performance under high user load.

## VI. Results and Discussion

The evaluation results confirm that real-time communication and dynamic ETA prediction significantly enhance the reliabil- ity and usability of public transport information systems. The modular Java full-stack architecture allows easy integration of additional features such as traffic analytics and machine learning-based prediction models.

The system is well-suited for deployment in smart city envi- ronments, benefiting both passengers and transport authorities.

## VII. Conclusion and Future Work

This paper presented a cloud-based real-time public trans- port tracking system developed using a Java full-stack archi- tecture. By integrating real-time data acquisition, WebSocket communication, and dynamic ETA prediction, the proposed system overcomes the limitations of traditional timetable- based transport systems.

Future work includes incorporating machine learning mod- els for traffic-aware ETA prediction, developing mobile appli- cations, and conducting large-scale deployment in real urban environments.

## References

1. S. Sonawane *et al.*, “Real-Time Bus Tracking System using GPS and GSM,” *International Journal of Engineering Research & Technology (IJERT)*, vol. 9, no. 6, pp. 1–5, 2020.
2. K. Ganesh, M. Thrivikraman, J. Kuri, H. Dagale, G. Sudhakar, and S. Sanyal, “Implementation of a Real Time Passenger Information System,” *arXiv preprint arXiv:1206.0447*, 2012.
3. H. Khaled, T. Wafeek, and H. Hussein, “Intelligent Transportation Sys- tem Real-Time Tracking,” *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 9, pp. 123–130, 2023.
4. A. B. P. Ashwini, R. Sumathi, and H. S. Sudhira, “A Dynamic Model for Bus Arrival Time

Estimation Based on Spatial Patterns Using Machine Learning," *arXiv preprint arXiv:2210.00733*, 2022.
5. F. Rashvand, S. Sanyal, and M. El-Bakry, "Real-Time Bus Departure Prediction Using Neural Networks for Smart IoT Public Bus Transit," *arXiv preprint arXiv:2501.10514*, 2025.
6. F. Rashvand, M. El-Bakry, and S. Sanyal, "Real-Time Bus Arrival Prediction: A Deep Learning Approach for Enhanced Urban Mobility," *arXiv preprint arXiv:2303.15495*, 2023.
7. N. Barnes, A. A. L. P. Oliveira, and K. Kenthapadi, "BusTr: Pre- dicting Bus Travel Times from Real-Time Traffic," *arXiv preprint arXiv:2007.00882*, 2020.
8. GTFS Realtime, "General Transit Feed Specification Realtime (GTFS- rt) Standard," Google Developers, Version 2.0, Mar. 2024. [Online]. Available: https://en.wikipedia.org/wiki/GTFS Realtime